accordance with 3G wireless network standards, the UE has four states: IDLE, CELL_FACH, CELL_DCH, and CELL_PCH. In the IDLE state, the UE is turned on, but an RRC connection has not yet been established. In this state, the UE consumes minimal energy. In CELL_DCH (dedicated channel), the UE is in an established state, and a dedicated channel is allocated by the network exclusively to the UE, which can be used for transferring uplink (UL) and downlink (DL) data. In this state, the UE consumes the most power. In CELL_FACH (forward access channel), the UE has established a connection with the network and the network has allocated shared channel resources to the UE. Finally, in CELL_PCH (paging channel), the UE consumes the lowest amount of current, and is unable to send or receive packets.

[0022] Referring now to FIG. **1**, an example of UE state transition in a 3G network is provided. For the UE to remain in the same state, the same data rate is to be maintained. For example, a UE **100** could start in the IDLE **102** state, where it consumes the least amount of power. When the UE **100** establishes a connection to the RAN, the UE can first move to CELL_FACH **104**, where a relatively low volume of data can be transmitted. When the UE **100** moves to a new application requiring a higher volume of data transmission, it moves to CELL_DCH **106**, where a dedicated channel is allocated exclusively to the UE. While the UE **100** is in CELL_DCH **106**, the highest amount of power is utilized. After a specified period of inactivity is met (based on a pre-configured inactivity timer, for example) and no data is exchanged, the UE **100** can then transition states again to either CELL_FACH **104** or CELL_PCH **108**, for example.

[0023] Utilizing a fixed or pre-configured timer in state transitions can be ineffective and reduce UE energy. Specifically, in some current RRC methods, there is a lack of guidance regarding applications' traffic type and duration, which can render the fixed timer configuration ineffective. For example, some operators have one set of configurations that is used during off-peak and on-peak hours, and that are homogeneously applied network-wide. While such a configuration may provide temporary relief in a crowded network, the fixed timers can lead to poor or inefficient RRC protocol because the same timer is utilized throughout the network, regardless of the load on the network or the traffic utilized by the application, for example.

[0024] The present disclosure addresses the above issues by implementing an active learning technique using an application server and enabling real-time adaptive timer management. Specifically, the present disclosure includes a method **200** in accordance with FIG. **2**. At **202**, application flows are analyzed at an application server (shown in FIG. **7** and described in greater detail below) with respect to at least one device connected to a network. At **204**, an adaptive timer value is generated at the application server based on the application flows of the at least one device. At **206**, the adaptive timer value is sent to at least one server, which then sends the adaptive timer value to the at least one device (**208**). At **210**, the at least one device adopts the adaptive timer value.

[0025] In the present disclosure, the application server is an IT server module known as a Radio Applications Cloud Server (RACS) **608** (described in further detail below with reference to FIG. **6**). The RACS is integrated into the RAN; for example, it can be integrated into the Radio Network Controller (RNC) in a 3G network or into the eNodeB (eNB) in an LTE network. The RACS enables, among other things, deployment and hosting of local applications at the RAN side in a virtualization computing environment and applying cloud technologies.

[0026] In the method **200**, the RACS analyzes the application flows with respect to the at least one device by utilizing a learning method **300**, which is illustrated in FIG. **3**. During the learning method, application behavior is analyzed, in that the RACS processes application flows and learns the IP flows that pass there through. The RACS also learns the application flows and the arrival rate of traffic with respect to each of the devices connected to the network. Broadly speaking, the analyzing or learning method includes, among other things, extracting information or input data from the at least one device. The extracted information can include, for example, IP flow information, associate request and response timing, subscriber information, device model information, and location information. In addition, during the learning method, the RACS analyzes, for example, packet processing, packet classification, request size, response size, flow identification, signature identification, and state transition detection. As will be described in further detail below, as a result of the learning method **300**, a learning model can be created to assist the RACS in identifying the adaptive timer value.

[0027] Referring specifically to FIG. **3**, in the learning method **300**, user plane traffic or application flows that flow through the RACS are taken as input data (**302**). Specifically, the RACS extracts the input data from the device(s) connected to the network. As indicated above, the extracted input data can be common protocol properties such as, for example, IP flow information, associate request and response timing, subscriber information, subscription information, service information, device model information, and other supplementary information as needed. At **304**, the RACS can communicate with a hypothesis history database (HHD). The HHD can store extracted device information from previous applications of the learning method **300**. The RACS can compare its values with those stored in the HHD and use the additional inputs in the HHD to improve the performance of the learning method **300** in both efficiency and accuracy, as will be described in further detail below. At **306**, the learning method **300** continues by analyzing additional data at the RACS, such as packet processing, packet classification based on key protocol extraction fields, the size of each request/response from/to each device on the network, flow identification, signature identification, state transition detection, burstiness detection, and time-of-day correlation, for example. However, the analyzing at **306** is not limited to these factors. The learning method **300** repeats steps **302-306** until an adaptive timer value can be determined to best suit the device. At **308**, the final output or adaptive timer value is communicated to the eNB or base transceiver station (BTS) for reconfiguring the network. Finally, at **310**, the adaptive timer value is communicated to the device, which then adopts the adaptive timer value.

[0028] The learning method **300** is also implemented within additional embodiments of the present disclosure. For example and turning now to FIG. **4**, a method **400** in accordance with the present disclosure is provided. In the method **400**, at least one device is connected to a network, and at **402**, initiates traffic on the network. For example, initiating traffic on the network can include initiating a web page request and/or using an application on the device,